

用 FOXPLUS 实现菜单系统设计

罗 升 陈洪森 (洪泽县工商支行) 张 群 (洪泽县医药公司)

弹出下拉式菜单程序设计是近年才兴起的一种人机友善的菜单设计方法;由于程序动向清晰、操作简便、功能交待明白、用户界面好,已得到了广泛的使用。如目前较普遍使用的 APPLE 公司的 Macintosh 操作系统、文字编辑软件 WPS、明星中文操作系统 ACIOS 中国龙等都采用弹出下拉式菜单程序设计。

FOXPLUS 作为 XENIX 及 DOS 操作系统下使用较广的数据库管理工具,本身并未为程序员提供弹出下拉式菜单功能或命令。如何才能使 FOXPLUS 数据库管理程序具有弹出下拉式菜单功能?笔者经过对 FOXPLUS 数据库系统的分析,通过系统本身提供的命令及函数经适当编程,较好地实现了弹出下拉式菜单功能。通过使用本程序执行效果较好;与其它的如 C 语言、PASCAL 语言等程序设计的弹出下拉式菜单功能及效果并无两样。现就根据 FOXPLUS 数据库系统本身的函数及命令处理,谈谈弹出下拉式菜单的一些程序设计技巧并提供相关程序。

大家都知道,FOXPLUS 数据库系统中有 DIME 设置数组语句及 INKEY() 键盘输入返回码函数;可以充分利用数组及 INKEY() 函数的功能,达到实现弹出下拉式菜单设计。具体方法为:首先将应用系统的操作系统在 CCDOS2.13I 以上版本支持,将具体需实现的功能以数组的形式保存起来,需要实现菜单时通过菜单专用程序将具体功能显示出来;菜单内容选择可通过键盘→、←、↑、↓、ESC 及回车键控制或通过键入相应数据得到(其中 INKEY() 函数对应有关键盘的返回 ASCII 码见 <附件一>)。程序采用模块结构设计,对菜单可根据实际情况灵活编制(本例程序设计为三级菜单形式)设计为多级菜单;各级菜单项数以数组形成存储,通过子模块的调用及返回。现将根据以上原理设计的有关程序附列如下(见<附件二>)。

<附件一>

INKEY()函数返回的键盘 ASCII 码

输入键盘码	inkey()返回码
→键	4
↑键	5
←键	19
↓键	24
←(回车)键	13
ESC 键	27

<附件二>

Y cat rsda.prg

** 总控菜单 BSDA.PRG

```

clca all
set safe off set scor off set echo off set talk OFF
SET STAT OFF set step off set esca on set dele on
DO WHIL .T.
set cons on
op = date()
op = dtoc(op)
set colo to
@1,1 say chr(14) + "[60]60@F * 15 人事档案]"
@1,1 say chr(14) + "[95]110@2423 * 14 信息系统]"
@1,1 say chr(14) + "CS]"
@1,1 say chr(14) + "D193,332]"
@1,1 say chr(14) + "B236,126]"
@1,1 say chr(14) + "C15]"
@1,1 say chr(14) + "D191,334]"
@1,1 say chr(14) + "B240,130]"
set colo to bg / n,bg / b
@12,25 say "主菜单版本 2.00"
set colo to w
@13,25 say "set colo to bg / n,bg / b
D1 = " 1.人事处理 "
D2 = " 2.系统维护 "
D3 = " 3.输出处理 "
D4 = " 0.退出系统 "
@14,25 say "1.人事处理 "
```

```

@15,25 say "2.系统维护"
@16,25 say "3.输出处理"
@17,25 say "0.退出系统"
hg=0
ttt=999
Y1=1
DDD='1'
SET COLO TO w+ / bg &&&&第一光条
@1,52 say op
do whil.t.
RRR=inkey() &&&&根据键盘输入返回 ASCII 码
DO CASE
CASE RRR=13 &&&&回车键将变量 Y1 带出 CASE
EXIT
CASE RRR=5 &&&&↓键
@13+Y1,25 SAY D&DDD
Y1=Y1-1
IF Y1=0
Y1=4
ENDI
DDD=STR(Y1,1)
SET COLO TO w+ / bg
@13+Y1,25 SAY D&DDD
set colo to bg / n,bg / b
CASE RRR=24 &&&&↑键
@13+Y1,25 SAY D&DDD
Y1=Y1+1
IF Y1=5
Y1=1
ENDI
DDD=STR(Y1,1)
SET COLO TO w+ / bg &&&&设置第一个光条
@13,Y1,25 SAY D&DDD
set colo to bg / n,bg / b
CASE CHR(RRR) >='0'.AND.CHR (RRR) <='8'
&&&&输入字符处理
Y1=VAL(CHR(RRR))
EXIT
ENDC
endd
IF Y1=4
Y1=0
ENDI
RR=CHR(ASC("Y1"))
set colo to
@1,1 say chr(14)+"R1]"
@1,1 say chr(14)+"B0]"
do case
case rr=0
@1,1 say chr(14)+"R1]"
clca prog
clos proc
@1,1 say chr(14)+"Q0]"
quit
case rr=3
clea prog
clos proc
clca all
exit
otherwise
do\rsda\prg\rs&RR &&&&调用二级菜单子模块
enddo
endd
retu
Y cat rs1.prg
* * 程序名:RS1.PRG
* * 功能:二级菜单控制
* M1,M2,M3,M4,M5,M6 三级菜单的个数
* TM 三级菜单长度
* FD 二级菜单间空
* 调用 MENU.PRG
elos all
pubiic mc
clca m1=1 m2=1 m3=1
m4=3 m5=5 m6=2 m7=4 ad1=0
dime me1(m1),me2(m2),me3(m3), me4(m4), me5 (m5), me6(m6),
me7(m7)
me1(1)="修改时间"
me2(1)='定期报表' &&&报表选择
me3(1)="单位选择"
me4(1)="数据录入"
me4(2)="文本接收"
me4(3)="平衡检查"
me5(1)="汇总报表" &&汇总分析
me5(2)="上报处理"
me5(3)="分析处理"
me5(4)=""
me5(5)=""
me6(1)="原始资料" &&打印
me6(2)="分析表"
me7(1)='通讯录'
me7(2)="计算器"
me7(3)="数据查询"
me7(4)="执行 DOS"
d1=1
d2=0
a1='报表时间'
a2="报表选择"
a3="单位"

```

```

a4="录入方式"
a5="汇总分析"
A6='表格打印'
A7='其他'
tm=len(a1)
hhh=1
me='1'
set colo to r+
@0,1 say '人事信息系统 V2.00'
set colo to gr+
@ 0,35 say '<<<业务处理>>>'
set colo to ng
@0,70 say date()
set colo to b / bg
fd=spac(1)
@1,0 say a1+fd+a2+fd+a3+fd+a4+fd+a5+fd+a6+fd+a7+spac(4)
&&&&显示首行菜单
fd=len(fd)
set colo to gr+ / rb,gr+ / bg &&&&光标定位于第一行第一条
@1,0 say a1
set colo to gr+ / bg,gr+ / b
kk=1000
do menu with '1',2,0,1+m1+2,tm-1,m1 &&&&调用三级菜单专门
处理程序,调用时带入六个参数
do while .t.
  xxfd1=d1
  xxfd2=d2
  xxffd=fd
  xxfhhh=hhh
if len(me)>1
  if me='74'
    run cd \rsda
    SET COLO TO gr+ / r
    @1,1 say chr(14)+"Q0]"
    @24,20 SAY "请输入 EXIT 按回车键返回人事系统"
    @1,1 say chr(14)+"r1]"
    run c:\command
    run cd \rsda\prg
    @1,1 say chr(14)+"r0]"
    retu to mast
  clsc
  set colo to
  @2,0 clea
  do \rsda \prg \rsl&mc
  ondl
d1=xxfd1
d2=xxfd2
fd=xxffd
hhh=xxfhhh

```

```

xx=224
kk=10
set colo to gr+ / bg,gr / bg
me='1'
endi
if kk=1000
  xx=inkey(1)
endi
if xx=0
  loop
endi
if kk=-1 &&&&子模块 MENU 返回 KK=-1 及
相应键为←
  xx=19
endi
if kk=1 &&&&子模块 MENU 返回 KK=1 及
相应键为→
  xx=4
endi
if xx=27 &&&&返回键码为回车←
  set colo to
  retu
endi
hfj=str(hhh,1)
do case
  case xx=4
    @d1,d2 say a&hfj
    if hhh=7
      hhh=1
      d2=0
    else
      hhh=hhh+1
    endi
    hfj=str(hhh,1)
    d2=(hhh-1)*(tm+fd)
    set colo to gr+ / rb,gr+ / bg
    @d1,d2 say a&hfj
    set colo to gr+ / b,gr+ / bg
    do menu with hfj,d1+1,d2, d1+2+m& hfj, d2+tm-1,
m&hfj
  loop
  case xx=224
    @d1,d2 say a&hfj
    hfj=str(hhh,1)
    set colo to gr+ / rb,gr+ / bg
    @d1,d2 say a&hfj
    set colo to gr+ / b,gr+ / bg
    do menu with hfj,d1+1,d2, d1+2+m & hfj, d2+tm-1,
m&hfj
  loop

```

```

case xx = 19
  @d1,d2 say a&hfj
  if hhh = 1
    hhh = 7
  clsc
    hhh = hhh-1
  endi
  hfj = str(hhh,1)
  d2 = (hhh-1) * (tm+fd)
  set colo to gr+ / rb,gr+ / bg
  @d1,d2 say a&hfj
  set colo to gr+ / b,gr+ / bg
  do menu with hfj,d1+1,d2, d1+2+m & hfj, d2+tm-1,
m&hfj
    loop
  endc
endd
set colo to
@2,0 elca
clos all
retu
  ¥ cat menu.prg

```

* 控制三级菜单程序

* MB 菜单数组下标

* AD1,AD2,AD3,AD4 坐标

* BA1 三级菜单个数

para mee,ad1,ad2,ad3,ad4,ba1 &&&&主调程序六个参数传递

set colo to gr+ / rb-

@ad1,ad2 clea to ad3-1,ad4

set colo to b / bg

@ad1,ad2+1 clea to ad3-2,ad4-1

f = 14ad1 = ad1-1

do while f <= ba1

@ad1+f,ad2+1 say me&mee(f)

f = f + 1

endd

set colo to gr+ / b,b / bg

@ad1+1,ad2+1,say me&mee(1)

set colo to b / bg

d = 1

hhh = 1

dx = ad1+1

hfj = str(hhh,1)

do while .t.

xx = inkey(1)

if xx = 27

&&&&ESC 键

kk = 1000

retu to mast

&&&&返回最高一级调用

endi

if xx = 0

loop

endi

if xx = 13

&&&&回车键

me = mee+str(hh,1)

kk = 1000

retu

endi

do case

case xx = 24

&&&&↓键

@dx,ad2+1 say mc&mcc(hh)

if hh = ha1

hh = 1

dx = ad1+1

elsec

hh = hh+1

dx = dx+1

endi

hfj = str(hh,1)

set colo to gr+ / b,b / bg

@dx,ad2+1 say me &mee(hh)

set colo to b / bg

case xx = 5

&&&&↑键

@dx,ad2+1 say me &mee(hh)

if hh = 1

hh = [m]

dx = ad1+1 * hh

else

hh = hh-1

dx = dx-1

endi

hfj = str(hh,1)

set colo to gr+ / b,b / bg

@dx,ad2+1 say me&mee(hh)

set colo to b / bg

case xx = 4

&&&&→键

kk = 1

set colo to

@ad1,ad2 elca to ad3,ad4

set colo to b / bg

retu

case xx = 19

&&&&←键

kk = -1

set colo to

@ad1,ad2 clea to ad3,ad4

set colo to b / bg

retu

endc

endd